

## Создание игры «Змейка» на JavaScript

*Семченко Регина Викторовна*

*Приамурский государственный университет имени Шолом-Алейхема*

*студент*

*Еровлев Павел Андреевич*

*Приамурский государственный университет имени Шолом-Алейхема*

*студент*

### Аннотация

В работе описаны возможности кода на JavaScript и разработка культовой игры змейка, известная еще со старых кнопочных телефонов. В ходе исследования произведен пошаговый обзор написания кода. Практическим результатом является рабочая игра.

**Ключевые слова:** браузер, JavaScript, змейка, игра

## Creating a «Snake» game in JavaScript

*Semchenko Regina Viktorovna*

*Sholom-Aleichem Priamursky State University*

*student*

*Erovlev Pavel Andreevich*

*Sholom-Aleichem Priamursky State University*

*student*

### Abstract

The paper describes the capabilities of JavaScript code and the development of the cult game snake, known since the old button phones. The study produced a step-by-step overview of code writing. The bottom line is a working game

**Keywords:** browser, javascript, snake, game

Snake – игра, которая известна большому количеству людей, ведь многие с ней сталкивались: в телефоне, на компьютерах, на консолях была эта игра. Суть игры простая, необходимо «змейкой» передвигаясь поедать «фрукты», один фрукт один балл и к тому же змейка прибавлялась в размерах, один фрукт на одно деление змейки больше. Игра заканчивалась, если «змейка» врезалась в границу карты, либо в саму себя. Игра ведется на счет, то есть кто больше заработал баллов, то и победил[1].

Цель данной статьи создать данную игру «змейка» на языке программирования Java Script и HTML.

При создании игры были изучены и прочитаны следующие источники. В учебном пособии В. В. Захаркина рассмотрены основы применения алгоритмического языка JavaScript при создании интерактивных веб-документов[4]. Т.В. Зудилова, М.Л. Буркова в своем пособии рассматривают необходимые для создания игры объекты `math` и `document`, а так же рассказывают о способе создания и вызова функций [5]. В своей работе Н. Прохоренко, В. Дронов рассмотрели вопросы создания интерактивных Web-сайтов с помощью HTML, JavaScript[8]. В работе С.А. Беляева рассматриваются ключевые вопросы разработки однопользовательских двумерных игр на языке JavaScript[3]. А.В. Диков исследовал объект `Math` языка JavaScript, который обладает множеством математических свойств и методов. Таким образом, открывается возможность использования языка веб-программирования для решения математических задач[12]

Для начала создадим папку «snake» и поместим в нее три папки «css», «js», «img», и HTML-файл, где пропишем страницу для нашей игры. При их написании использовались рекомендации из источников [10-11].

HTML-файл представлен ниже (рис.1).

```
<!DOCTYPE html>
<html lang="ru">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta http-equiv="X-UA-Compatible" content="ie=edge">
<title>Snake</title>
<link rel="stylesheet" href="css/style.css">
</head>
<body>
<canvas id="game" width="608" height="608"></canvas>
<script src="js/game.js"></script>
</body>
</html>
```

Рисунок 1 - Содержимое HTML-файла

Здесь была создана начальная web-страница с подключенным css и js, так же были заданы границы для поля игры. Это все, что будет находится в этом файле, следующая задача создать скрипт позволяющий работать игре.

Теперь в таблице стилей необходимо выровнять поле, что бы оно стояло ровно по центру.

В папке css создадим файл `style.css` и пропишем пару строчек (рис. 2).

```
canvas {
  display: block;
  margin: 0 auto;
}
```

Рисунок 2 - Файл `style.css`

Теперь осталось добавить в папку `img` фотографии поля и еды, после чего можно приступать к написанию Java-кода. Картинки можно найти на сайте `ICONFINDER`, в разделе бесплатные и скачать в нужном размере[2]. Поле должно быть 608 на 608 пикселей т.к. был указан размер нашего поля (рис.3). Картинка поля разделена на множество блоков, где каждый блок 32 на 32 пикселя, соответственно необходима картинка еды под такие размеры (рис.4).

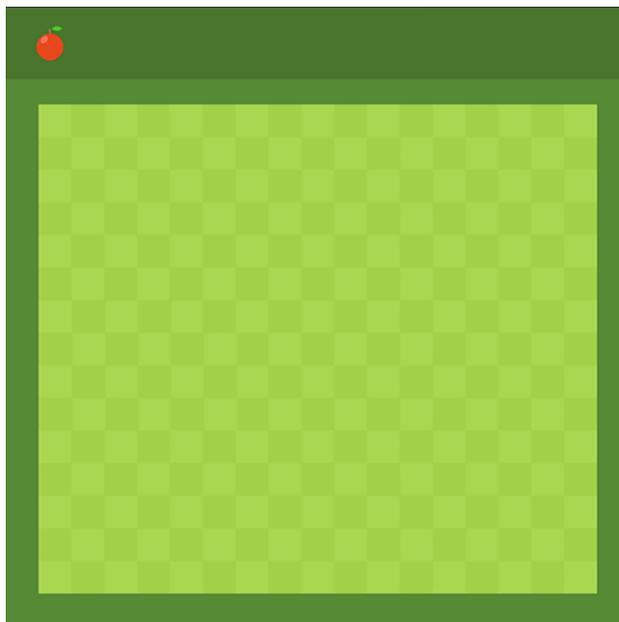


Рисунок 3 – Поле игры



Рисунок 4 – Еда

После того как все подготовительные работы сделаны, можно приступать к написанию скрипта. Создадим в папке `js` файл с названием `game.js`.

В файле напишем, что игры будет 2d, добавим картинки поля и еды, обозначим начальный счет и размер еды, это будем началом (рис.5).

```
const canvas = document.getElementById(«game»);
const ctx = canvas.getContext("2d");
const ground = new Image();
ground.src = "img/ground.png";

const foodImg = new Image();
foodImg.src = "img/food.png";

let box = 32;
let score = 0;
```

Рисунок 5 - Код JS

Дальше напишем, чтобы еда появлялась в случайном местоположении, а так же чтоб сама змейка имела начальные координаты ровно в центре (рис.6).

```
let food = {
x: Math.floor((Math.random() * 17 + 1)) * box,
y: Math.floor((Math.random() * 15 + 3)) * box,
};

let snake = [];
snake[0] = {
x: 9 * box,
y: 10 * box
};
```

Рисунок 6 - Назначение координат

Следующим шагом будет написание функций, пропишем способ управления, если «змейка» зайдет на границу поля, то игра останавливается, если она съест сама себя, то игра останавливается, пропишем количество очков за фрукт, а так же пропишем, если «змейка» съедает фрукт, то она увеличивается в размерах, в нашем случае это будут квадраты разных цветов, цвет головы будет черный, а цвет туловища желтый.

Ниже представлен весь код JS (рис.7).

```
const canvas = document.getElementById("game");
const ctx = canvas.getContext("2d");
const ground = new Image();
ground.src = "img/ground.png";

const foodImg = new Image();
foodImg.src = "img/food.png";

let box = 32;
let score = 0;
let food = {
x: Math.floor((Math.random() * 17 + 1)) * box,
y: Math.floor((Math.random() * 15 + 3)) * box,
};

let snake = [];
snake[0] = {
x: 9 * box,
y: 10 * box
};

document.addEventListener("keydown", direction);
let dir;
function direction(event) {
if(event.keyCode == 37 && dir != "right")
dir = "left";
else if(event.keyCode == 38 && dir != "down")
dir = "up";
else if(event.keyCode == 39 && dir != "left")
```

```
dir = "right";
else if(event.keyCode == 40 && dir != "up")
dir = "down";
}

function eatTail(head, arr) {
for(let i = 0; i < arr.length; i++) {
if(head.x == arr[i].x && head.y == arr[i].y)
clearInterval(game);
}
}

function drawGame() {
ctx.drawImage(ground, 0, 0);
ctx.drawImage(foodImg, food.x, food.y);
for(let i = 0; i < snake.length; i++) {
ctx.fillStyle = i == 0 ? "black" : "yellow";
ctx.fillRect(snake[i].x, snake[i].y, box, box);
}

ctx.fillStyle = "white";
ctx.font = "50px Arial";
ctx.fillText(score, box * 2.5, box * 1.7);

let snakeX = snake[0].x;
let snakeY = snake[0].y;

if(snakeX == food.x && snakeY == food.y) {
score++;
food = {
x: Math.floor((Math.random() * 17 + 1)) * box,
y: Math.floor((Math.random() * 15 + 3)) * box,
};
} else
snake.pop();
if(snakeX < box || snakeX > box * 17
|| snakeY < 3 * box || snakeY > box * 17)
clearInterval(game);
if(dir == "left") snakeX -= box;
if(dir == "right") snakeX += box;
if(dir == "up") snakeY -= box;
if(dir == "down") snakeY += box;

let newHead = {
x: snakeX,
y: snakeY
};

eatTail(newHead, snake);
snake.unshift(newHead);
}

let game = setInterval(drawGame, 100);
```

Рисунок 7 - Код игры

Теперь запустив HTML-файл выйдет окно с игрой, управляя на стрелочках на клавиатуре, можно поиграть в эту незабываемую игру (рис.8).

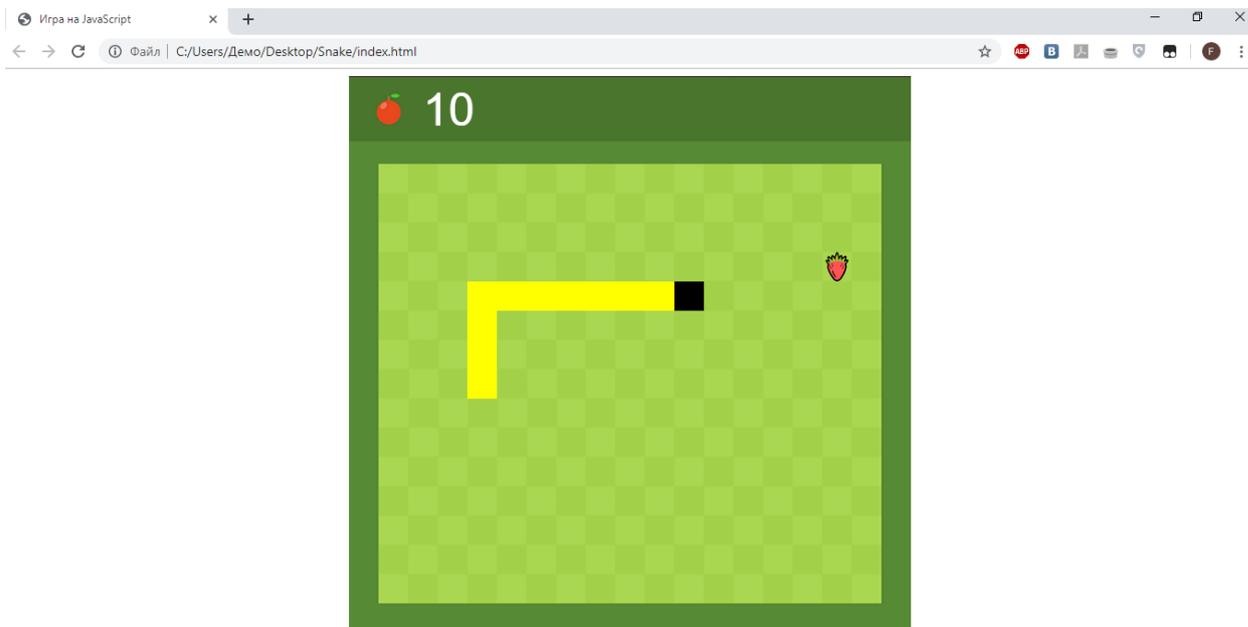


Рисунок 8 – Игра змейка

Таким образом, реализация игры «змейка» на языке Java возможна и с помощью несложного кода можно поиграть в собственную игру. В результате исследования была произведена пошаговая реализация проекта по разработке браузерной игры змейка. Практическим результатом получается рабочая браузерная игра.

### Библиографический список

1. Семченко Р.В , Еровлев П.А Создание змейки на C++ // Постулат. 2019. №6. URL: <http://e-postulat.ru/index.php/Postulat/article/view/2662/2706>
2. ICONFINDER URL: <https://www.iconfinder.com/> (дата обращения: 20.08.2019).
3. Беляев С.А. Разработка игр на языке JavaScript: Учебное пособие. СПб.: Лань, 2016. 128с.
4. Захаркина В. В. JavaScript. Основы клиентского программирования: Учебное пособие. СПб.: Ф-т филологии и искусств СПбГУ, 2007. 73 с.
5. Зудилова Т.В., Буркова М.Л Web-программирование JavaScript. СПб: НИУ ИТМО, 2012. 68 с.
6. Что такое JavaScript? Основные функции JavaScript. [http://www.progaprosto.ru/doc/yazyk\\_programmirovaniya\\_javascript.php](http://www.progaprosto.ru/doc/yazyk_programmirovaniya_javascript.php)
7. Основы Javascript. URL: <https://learn.javascript.ru/first-steps> (дата обращения 20.08.2018)
8. Прохоренок Н., Дронов В. HTML, JavaScript, PHP и MySQL. СПб.: БХВПетербург, 2015. 766 с.

9. Флэнаган Д. JavaScript. Подробное руководство. М.: Символ-Плюс, 2012. 357с.
10. Язык программирования JavaScript. URL: [http://progaprosto.ru/doc/yazyk\\_programmirovaniya\\_javascript.php](http://progaprosto.ru/doc/yazyk_programmirovaniya_javascript.php) (дата обращения 20.08.2018).
11. Wiki-учебник по веб-технологиям: JavaScript. URL: <http://www.webmasterwiki.ru/JavaScript> (дата обращения 20.08.2018).
12. Диков А.В. Математические алгоритмы на JavaScript // Известия Пензенского государственного педагогического университета им. В.Г. Белинского. 2009. № 17. С. 84-88.